МОДЕЛИРОВАНИЕ АТАК ДЛЯ АКТИВНОГО АНАЛИЗА УЯЗВИМОСТЕЙ КОМПЬЮТЕРНЫХ СЕТЕЙ

М. В. Степашкин, И. В. Котенко, В. С. Богданов (Санкт-Петербург)

Введение

Согласно исследованиям CERT [1] количество атак на компьютерные сети (КС), их сложность и размер ущерба, вызванного злоумышленными атаками в сети Интернет, растет с каждым годом. Причиной этого является низкий уровень защищенности (УЗ) большинства систем, подключенных к сети Интернет. Поэтому в настоящее время актуальной задачей является обнаружение уязвимостей и оценка уровня защищенности КС. Для решения данной задачи служит специальный класс систем, называемых системами анализа защищенности (САЗ) [15].

Данная работа посвящена разработке моделей, архитектур и прототипов интеллектуальных компонентов анализа уязвимостей и определения УЗ на основе моделирования атак, которые позволят расширить функциональные возможности существующих САЗ за счет имитации компьютерных атак (действий злоумышленника). В работе представлена архитектура САЗ и реализованных в ней моделей.

Обзор существующих работ

Основные подходы к анализу уязвимостей и оценке уровня защищенности базируются на аналитических вычислениях и имитационном моделировании. Аналитические подходы, как правило, используют различные методы оценки рисков [2, и др.]. Методы имитационного моделирования базируются на моделировании компьютерной сети, деревьях атак, моделей графов и др. [9, 10, 14 и др.].

В настоящее время существует много работ, раскрывающих различные подходы к моделированию атак: сети Петри [13], метод анализа изменения состояний [11], эмуляция вторжений в последовательном и параллельном режимах [5], причинно-следственная модель [6], концептуальные модели компьютерных вторжений [21], описательные модели сети и злоумышленников, структурированное описание на базе деревьев [7], моделирование “выживания” компьютерных систем [16], объектно-ориентированное дискретное событийное моделирование [3], модель запрос/ответ для компьютерных атак [22], ситуационное исчисление и целенаправленный вызов процедур [8], использование графов атак для анализа уязвимостей [12, 18] и т.д.

На этапе эксплуатации компьютерных систем для анализа уязвимостей и определения уровня защищенности могут использоваться две основные группы методов: пассивные (анализ журналов регистрации событий и т.п.) и активные (тестирование на проникновение) [4, 17]. Существует множество САЗ, функционирующих на этапе эксплуатации, например, Retina, Internet Scanner, CyberCop Scanner, Nessus Security Scanner и т.д. Их основными недостатками являются: 1) отсутствие ответа на вопрос “Какие ошибки в политике безопасности были обнаружены в процессе сканирования?”; 2) использование активного анализа уязвимостей для функционирующей системы может привести к нарушению работоспособности отдельного сервиса или системы в целом и т.д.

Архитектура системы анализа защищенности

Архитектура предлагаемой САЗ состоит из следующих модулей (рис. 1): 1) интерфейса пользователя; 2) реализации модели злоумышленника; 3) генерации комплекса сценариев; 4) выполнения сценария; 5) хранилище данных и знаний; 6) обновления

баз данных и знаний; 7) оценки уровня защищенности; 8) генерации отчетов; 9) сетевой интерфейс.

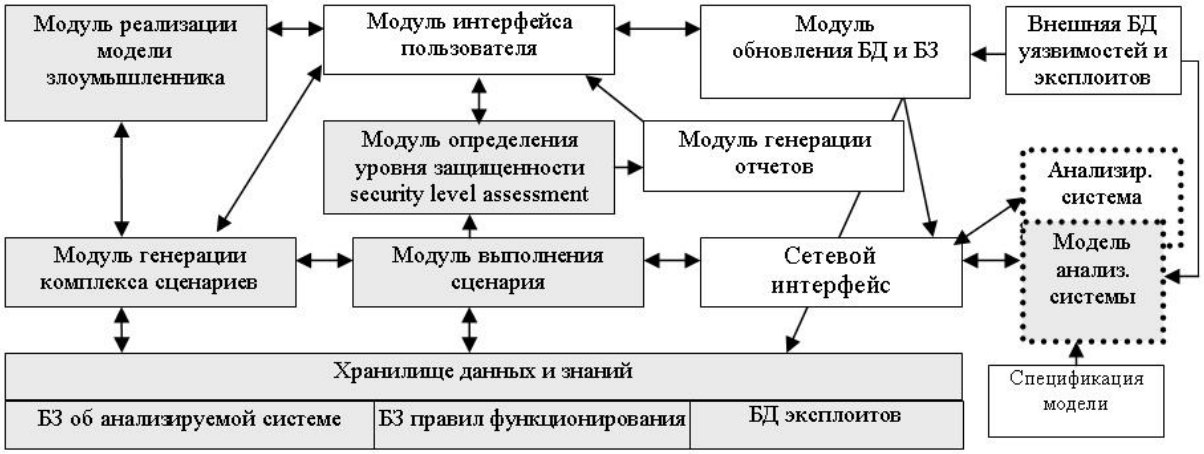

Рис. 1. Обобщенная архитектура системы анализа защищенности

На этапе проектирования КС САЗ оперирует с моделью сети, которая базируется на заданной спецификации. На этапе эксплуатации КС САЗ взаимодействует с реальной сетью (системой).

Обобщенная модель атак

Функционирование САЗ определяется моделью атак, которая имеет вид иерархической структуры, состоящей из нескольких уровней (рис. 2). Верхними уровнями являются комплексный (определяет множество высокоуровневых целей), сценарный (определяет одну высокоуровневую цель) и уровень этапов сценария. Множество этапов сценария состоит из разведки, внедрения, повышения привилегий; реализации угрозы; сокрытия следов; создания потайных ходов. Нижние уровни служат для уточнения злоумышленником подцелей атаки. Последний уровень в иерархии описывает низкоуровневые действия злоумышленника (запуск эксплоитов).

В модели атак используются два метода достижения злоумышленником конечной цели: 1) прямой и 2) обратный вывод. Оба метода используют базу правил функционирования САЗ для выбора узла иерархии обобщенной модели атак. При реализации стратегии прямого вывода САЗ выполняет все (или ограниченное количество) доступные на текущем уровне иерархии низкоуровневые действия злоумышленника для каждого этапа сценария, начиная с первого. Обратный вывод подразумевает создание оптимизированной цепочки действий на базе заданной злоумышленником конечной цели, начиная с последнего действия в цепочке и заканчивая первым.

Стратегия поведения злоумышленника задается его моделью: злоумышленник с низким уровнем умений использует метод прямого вывода для достижения цели; с высоким уровнем умений – обратного.

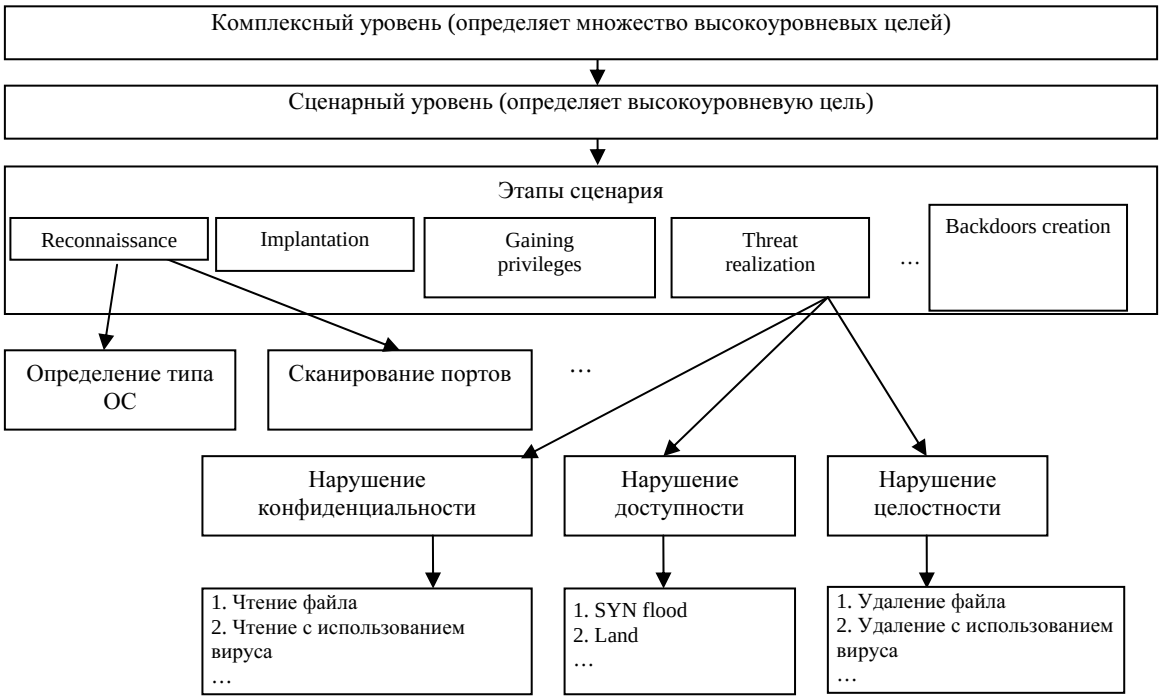

Рис. 2. Фрагмент обобщенной модели атак

Модель анализируемой компьютерной сети

Модель анализируемой КС служит для оценки результата атак и определения реакции сети на них. Она состоит из следующих основных модулей (рис.3): распознавания действий злоумышленника; вычисления результата выполнения атак; генерации откликов системы; базы знаний об анализируемой сети (системы); базы данных сигнатур атак; сетевого интерфейса.

Одним из основных модулей является модуль вычисления результата атак, использующий множество правил, которые описывают, какой тип атаки, при каких условиях и с какой вероятностью успеха реализуется. Входными данным для правил являются идентификатор атаки и множество параметров, описывающих текущее состояние анализируемой системы. Выходным значением является вероятность успешного выполнения атаки.

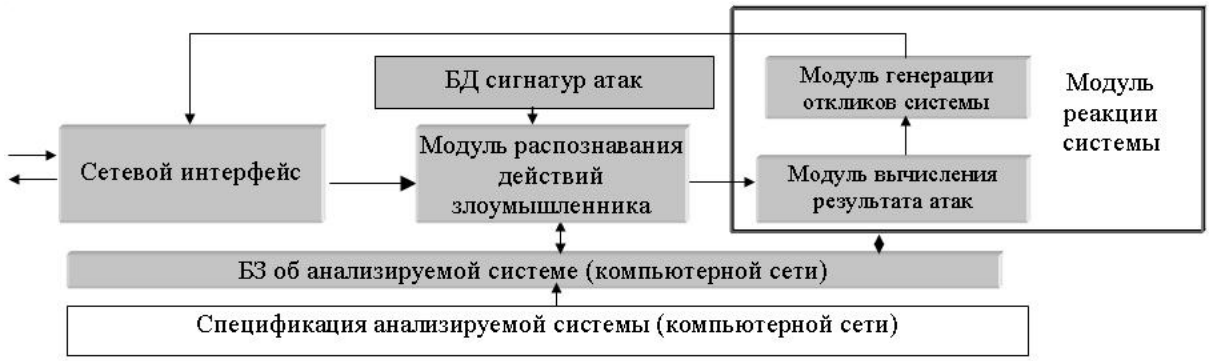

Рис. 3. Модель анализируемой компьютерной сети

Реакцией модели информационной системы на имитируемые атаки является изменение ее состояния и (в определенных случаях) некоторое сообщение злоумышленнику. Каждое состояние системы характеризуется следующими основными параметрами: 1) доступность системы в целом и ее отдельных сервисов; 2) целостность данных; 3) конфиденциальность данных; 4) пользователи и их права и т.д. Модуль генерации откликов системы использует следующее множество правил: $\{R^{SR}_j\}$, где R^{SR}_j: *Input -> Output & Post-Condition*, *Input*- действия злоумышленника, *Output* – отклик системы, *Post-Condition* – изменение состояния системы, & – логическое "И".

Модель определения уровня защищенности

Модель определения уровня защищенности базируется на функционировании *модуля определения уровня защищенности*, основной задачей которого является расчет метрик безопасности (МБ). Множество метрик образует таксономию МБ, которая содержит понятия действий по реализации атак, а также понятия типов и категорий объектов защиты (ОЗ). В таксономии метрик безопасности на основе атакующих действий выделим четыре уровня: 1) интегральный; 2) сценарный; 3) этапов сценария; 4) реализации угрозы. Каждый вышележащий уровень содержит все метрики нижележащих уровней. Примером метрики данной таксономии является метрика "Общее количество и количество успешно выполненных сценариев". Вторая таксономия базируется на типах (информационные ресурсы, программные ресурсы, физические ресурсы, сервисы) и категориях (по уровню конфиденциальности и критичности) ОЗ. Примером метрики данной таксономии является метрика "Общий уровень конфиденциальности и критичности успешно атакованных ОЗ".

Определение МБ основывается на информации о результатах атак и реакции анализируемой системы на них.

Тестовая компьютерная сеть

Для тестирования и оценки предлагаемого подхода мы специфицировали и реализовали компьютерную сеть, состоящую из трех подсетей: 1) части глобальной сети Интернет; 2) демилитаризованной зоны (ДМЗ); 3) локальной вычислительной сети (ЛВС). Основными элементами тестовой компьютерной сети являются: 1) Internet_host, на котором функционирует САЗ; 2) Firewall_1 – межсетевой экран между глобальной сетью Интернет и ДМЗ; 3) FTP-сервер, 4) почтовый сервер и 5) web-сервер, расположенные в ДМЗ; 6) Firewall_2 – межсетевой экран между локальной вычислительной сетью и демилитаризованной зоной; 7) DC – контроллер домена; 8) Workstation 1..4 – рабочие станции ЛВС.

Модель анализируемой системы использует спецификации политик безопасности и архитектуры данной компьютерной сети, заданные на специализированных языках Security Policy Language и System Description Language.

Выводы

В работе предложен имитационный подход к анализу уязвимостей и оценке защищенности КС. Основными компонентами предлагаемой САЗ является база правил функционирования, модель компьютерных атак и модель оценки уровня защищенности с использованием разработанной таксономии метрик безопасности. С использованием тестовой компьютерной сети был реализован прототип САЗ и проведены эксперименты. Направлениями дальнейших исследований является совершенствование моделей компьютерных атак и оценки уровня защищенности и проведение экспериментальной оценки предложенных решений.

Работа выполнена при финансовой поддержке РФФИ (проект №04-01-00167), программы фундаментальных исследований ОИТВС РАН (контракт №3.2/03) и при частичной финансовой поддержке, осуществляемой в рамках проекта Евросоюза POSITIF (контракт IST-2002-002314).

Литература

1. CERT/CC Statistics 1988-2005. http://www.cert.org/stats/cert_stats.html.
2. **Chapman C., Ward S.** Project Risk Management: processes, techniques and insights. Chichester, John Wiley, 2003.
3. **Chi S.-D., Park J.S., Jung K.-C., Lee J.-S.** Network Security Modeling and Cyber Attack Simulation Methodology//LNCS. –2001. Vol. 2119.
4. **Chirillo J.** Hack Attacks Testing – How to Conduct Your Own Security Audit. Wiley Pub-lishing, 2003.
5. **Chung M, Mukherjee B., Olsson R. A., Puketza N.** Simulating Concurrent Intrusions for Testing Intrusion Detection Systems//Proc. of the 18th NISSC, 1995.
6. **Cohen F.** Simulating Cyber Attacks, Defenses, and Consequences//IEEE Symposium on Security and Privacy. – Berkeley, CA, 1999.
7. **Dawkins J., Campbell C., Hale J.** Modeling network attacks: Extending the attack tree paradigm. Workshop on Statistical and Machine Learning Techniques in Computer Intrusion Detection, Johns Hopkins University, 2002.
8. **Goldman R. P.** A Stochastic Model for Intrusions//LNCS. – 2002. Vol. 2516.
9. **Gorodetsk, V., Kotenko I.** Attacks against Computer Network: Formal Grammar-based Framework and Simulation Tool. RAID 2000//LNCS. – 2002. Vol. 2516.
10. **Hariri S., Qu G., Dharmagadda T., Ramkishore M., Raghavendra C. S.** Impact Analysis of Faults and Attacks in Large-Scale Networks//IEEE Security & Privacy, Septem-ber/October, 2003.
11. **Iglun K., Kemmerer R. A., Porras P. A.** State Transition Analysis: A Rule-Based Intrusion Detection System//IEEE Transactions on Software Engineering, 21(3), 1995.
12. **Jha S., Sheyner O., Wing J.** Minimization and reliability analysis of attack graphs. Technical Report CMU-CS-02-109, Carnegie Mellon University, 2002.
13. **Kumar S., Spafford E. H.** An Application of Pattern Matching in Intrusion Detection. Technical Report CSDTR 94 013. Purdue University, 1994.
14. **Lye K., Wing J.** Game Strategies in Network Security//International Journal of Information Security. – 2005. – February.
15. **McNab C.** Network Security Assessment. O'Reilly Media, Inc., (2004).
16. **Moitra S. D., Konda S. L.** A Simulation Model for Managing Survivability of Networked Information Systems, Technical Report CMU/SEI-2000-TR-020, December, 2000.
17. Nessus Network Auditing. Renaud Deraison. Syngress Publishing, Inc., 2004.
18. **Ortalo R., Dewarte Y., Kaaniche M.** Experimenting with quantitative evaluation tools for monitoring operational security//IEEE Trans. on Software Engineering, 25(5), 1999.
19. OSVDB: The Open Source Vulnerability Database. http://www.osvdb.org/
20. **Rohse M.** Vulnerability naming schemes and description languages: CVE, Bugtraq, AVDL and VulnXML. SANS GSEC PRACTICAL. – 2003.
21. **Stewart A. J.** Distributed Metastasis: A Computer Network Penetration Methodology// Phrack Magazine., – 1999. – 9 (55).
22. **Templeton S. J., Levitt K. A** Requires/Provides Model for Computer Attacks. Proc. of the New Security Paradigms Workshop, 2000.